

IN THE CLAIMS:

1. (Previously presented) A computer implemented process for applying a set of rules, the process comprising:
 - (a) placing a pre-method control point before logic of a method and post-method control point after the logic of the method;
 - (b) associating a set of rules with each control point based on a class of object in which the method resides, name of the method, and type of control point, whether the pre-method control point or the post-method control point;
 - (c) invoking the method, wherein encountering each control point during the execution of the method comprises:
 - (i) determining if the encountered control point is active;
 - (ii) on the basis of an active control point:
 - 1) selecting rules based on a set of rules associated with the active control point associated in step (b);
 - 2) running the selected rules;
 - 3) obtaining results from running the rules; and
 - 4) combining the results using a combining algorithm specified by the control point.
2. (Previously presented) A computer implemented process for applying a set of rules comprising:
 - (a) defining an object;
 - (b) defining at least one method in the object;
 - (c) defining a control point just before logic of at least one method, wherein the control point is a method context control point; and
 - (d) associating a set of rules with the control point.
3. (Original) In the process of claim 2, the step of defining a first control point further comprises:
 - (a1) decorating the object to dynamically insert a first control point such that

the object acquires this new control point.

4. (Original) In the process of claim 2, the step of defining at least one control point further comprises:

(c1) adding the at least one control point through the technique of generating required code in the compiler or with a preprocessor.

5. (Original) In the process of claim 2, the step of defining at least one control point further comprises:

(c1) manually inserting the at least one control point and encoding the control point in the implementation of a hosting object.

6. (Original) In the process of claim 2, the step of defining at least one control point further comprises:

(c1) externalizing the at least one control point as a class and instantiating it at the at least one control point.

7. (Original) The process of claim 2 further comprises:

(e) defining a second control point just after the logic of each method; and

(f) associating a second set of rules with the second control point.

8. (Original) In the process of claim 7, wherein the rules in the second set of rules are associated to the second control point without considering the rules in the first set of rules associated with the at least one control point.

9. (Original) In the process of claim 7, wherein a set of rules is defined as having N number of rules, N being at least zero.

10. (Original) In the process of claim 2, the step of associating at least one control point further comprises:

(c1) defining, with a control point, at least one of a rule selecting algorithm and

a rule-results combination algorithm.

11. (Original) The process of claim 2, further comprises:
- (e) changing rules associated with the control point contained in the set of rules.
12. (Previously presented) A computer implemented process for applying a set of rules, comprising:
- (a) invoking a method in an object;
 - (b) encountering an active control point during the invocation of the method, wherein the active control point is an active method context control point;
 - (c) selecting rules associated with the method of the object at the control point;
 - (d) invoking the rules; and
 - (e) combining results from invoking the rules.
13. (Original) The process of claim 12, wherein the rules perform a variety of actions conditioned by the fact that rules may be associated with particular, regularly occurring points in the object model.
14. (Original) The process of claim 12, wherein the rules perform at least one function which varies over time.
15. (Original) A process of claim 12, wherein a control point occurs just before logic of the method begins, just after the logic of the method completes, or at both just before logic of the method begins and just after the logic of the method completes.
- 16-18. (Cancelled)
19. (Currently amended) ~~In the process of claim 18, the step of affecting the behavior of the object further comprises~~ A computer implemented process for applying a set of

rules comprising:

- (a) defining an object;
- (b) defining at least one method in the object;
- (c) defining at least one control point in the at least one method;
- (d) defining rules to the at least one control point on basis the object's class name, method, name, and position of the at least one control point in the method;
- (e) encountering a first control point;
- (f) running the rules associated with the first control point
- (g) affecting behavior of the object base on running the rules associated with the first control point; and
- (h) associating different rules to a control point.

20. (Currently amended) ~~In the process of claim 18, the step of affecting the behavior of the object further comprises~~ A computer implemented process for applying a set of rules comprising:

- (a) defining an object;
- (b) defining at least one method in the object;
- (c) defining at least one control point in the at least one method;
- (d) defining rules to the at least one control point on basis the object's class name, method, name, and position of the at least one control point in the method;
- (e) encountering a first control point;
- (f) running the rules associated with the first control point;
- (g) affecting behavior of the object base on running the rules associated with the first control point; and
- (h) defining another control point.

21. (Currently amended) ~~In the process of claim 18, the step of affecting the behavior of the object further comprises~~ A computer implemented process for applying a set of rules comprising:

- (a) defining an object;
- (b) defining at least one method in the object;

- (c) defining at least one control point in the at least one method;
- (d) defining rules to the at least one control point on basis the object's class name, method, name, and position of the at least one control point in the method;
- (e) encountering a first control point;
- (f) running the rules associated with the first control point;
- (g) affecting behavior of the object base on running the rules associated with the first control point; and
- (h) associating rules to a second control point.

22. (Cancelled)

23. (Previously presented) A computer implemented process for applying a set of rules, comprising:

- (a) defining an object;
- (b) defining a method in the object;
- (c) defining a first control point of the method, the first control point being a method context control point;
- (d) determining rules associated with the first control point;
- (e) defining a second control point of the method, the second control point also being a method context control point; and
- (f) determining rules associated with the second control point.

24. (Original) A computer implemented process as in claim 23 further comprising:

- (g) separately selecting, running and combining the results of rules determined to be associated with either control point.

25. (Original) In the process of claim 23 wherein the first control point is a pre-method trigger point.

26. (Original) In the process of claim 23 wherein the second control point is a post-method trigger point.

27. (Previously presented) A computer implemented process for defining an object comprising:

- defining an object;
- defining a method in the object by:
 - defining method logic;
 - placing the method logic in the method;
 - defining at least one control point, wherein the at least one control point is a method context control point; and
 - placing the at least one control point in the method wherein the method logic is continuous, wherein the step of placing the at least one control point further comprises placing the at least one control in the method after the method logic.

28. (Original) A computer implemented process for defining an object as in claim 27, wherein the step of placing the at least one control point further comprises placing the at least one control in the method before the method logic.

29-30. (Cancelled)

31. (Original) A computer implemented process for defining an object as in claim 27, further comprises:

- flagging the at least one control point on the basis of being active.

32. (Original) A computer implemented process for defining an object as in claim 27, wherein the step of defining the at least one control point further comprising:

- defining a rule selection algorithm associated with the at least one control point.

33. (Original) A computer implemented process for defining an object as in claim 27, wherein the step of defining the at least one control point further comprising:

- defining a rule result combination algorithm associated with the at least one control point.

34. (Original) A computer implemented process for defining an object as in claim 27, wherein the step of defining the at least one control point further comprises:
defining a rule selection algorithm for the at least one control point; and defining a rule result combination algorithm for the at least one control point.
35. (Original) A computer implemented process for defining an object as in claim 27, further comprising:
associating at least one rule with the at least one control point.
36. (Previously presented) A computer implemented process for defining a rule comprising:
creating the rule;
associating the rule with an object class;
associating the rule with a method within the object class;
associating the rule with an occurrence of a control point within the method; and
associating the rule with another method within the object class.
37. (Original) A computer implemented process for defining a rule as in claim 36 wherein the occurrence of the control point within the method being before method logic.
38. (Original) A computer implemented process for defining a rule as in claim 36 wherein the occurrence of control point within the method being after method logic.
39. (Original) A computer implemented process for defining a rule as in claim 36, further comprising:
associating the rule with another object class.
40. (Cancelled)

41. (Original) A computer implemented process for defining a rule as in claim 36, further comprising:
 associating the rule with another control point within the method of the object class.
42. (Original) A computer implemented process for applying a set of rules, comprising:
 selecting an object class;
 selecting a method within the object class;
 invoking the method;
 processing rules associated with the method comprising:
 encountering a control point associated with the method;
 determining if the control point is active; and
 finding at least one rule associated with an active control point.
43. (Original) A computer implemented process for applying a set of rules as in claim 42, wherein the step of finding at least one rule further comprises:
 accessing a selecting algorithm associated with the active control point; and
 selecting at least one rule using the selecting algorithm.
44. (Original) A computer implemented process for applying a set of rules as in claim 42, where in the step of processing rules further comprises:
 running the at least one rule;
 determining results from running the at least one rule;
 accessing a combining algorithm associated with the control point; and
 combining the results using the combining algorithm.
45. (Original) A computer implemented process for applying a set of rules, comprising:
 selecting an object class;
 selecting a method within the object class;

- invoking the method;
- processing rules comprising:
- encountering a control point;
- accessing a selecting algorithm associated with the control point; and
- selecting at least one rule using the selecting algorithm.

46. (Previously presented) A computer implemented process for applying a set of rules, comprising:

- selecting an object class;
- selecting a method within the object class;
- invoking the method;
- processing rules comprising:
 - encountering a method context control point;
 - finding at least one rule associated with the method context control point;
 - running the at least one rule;
 - determining results on the basis of running the at least one rule;
 - accessing a combining algorithm associated with the method context control point; and
 - combining the results using the combining algorithm.

47. (Original) A computer implemented process for applying a set of rules, comprising:

- selecting an object class;
- selecting a method within the object class;
- invoking the method;
- processing rules comprising:
 - encountering a first control point associated with the method;
 - determining if the first control point is active;
 - executing method logic of the method;
 - encountering a second control point associated with the method;
 - determining if the second control point is active; and

finding a set of rules associated with one of the first control point and the second control point, wherein the set of rules contains not less than zero rules.

48. (Previously presented) A computer implemented process for applying a set of rules, comprising:

- selecting an object class;

- selecting a method within the object class;

- invoking the method;

- processing rules comprising:

 - encountering a control point associated with the method, the control point being a method context control point for which associated rules may be changed from a first set of rules to a second set of rules different from the first set of rules;

 - finding at least one rule associated with the control point prior to executing method logic of the method;

 - running the at least one rule;

 - obtaining results on the basis of running the at least one rule; and

 - controlling the method on the basis of the results.

49. (Original) A computer implemented process for applying a set of rules as in claim 48, wherein the step of controlling the method comprises:

- exiting the method.

50. (Original) A computer implemented process for applying a set of rules as in claim 48, wherein the step of controlling the method comprises:

- executing method logic of the method.

51. (Previously presented) A data processing system for defining an object comprising:

- defining means for defining an object;

- defining means for defining a method in the object by:

- defining means for defining method logic;

placing means for placing the method logic in the method;

defining means for defining at least one control point, wherein the at least one control point is a method context control point; and

placing means for placing the at least one control point in the method wherein the method logic is continuous, wherein the step of placing the at least one control point further comprises placing means for placing the at least one control in the method after the method logic.

52. (Original) A data processing system for defining an object as in claim 51, wherein the step of placing the at least one control point further comprises placing means for placing the at least one control in the method before the method logic.

53-54. (Cancelled)

55. (Original) A data processing system for defining an object as in claim 51, further comprises:

flagging means for flagging the at least one control point on the basis of being active.

56. (Original) A data processing system for defining an object as in claim 51, wherein the step of defining the at least one control point further comprising:

defining means for defining a rule selection algorithm associated with the at least one control point.

57. (Original) A data processing system for defining an object as in claim 51, wherein the step of defining the at least one control point further comprising:

defining means for defining a rule result combination algorithm associated with the at least one control point.

58. (Original) A data processing system for defining an object as in claim 51, wherein the step of defining the at least one control point further comprises:

defining means for defining a rule selection algorithm for the at least one control point; and

defining a rule result combination algorithm for the at least one control point.

59. (Original) A data processing system for defining an object as in claim 51, further comprising:

associating means for associating at least one rule with the at least one control point.

60. (Previously presented) A data processing system for defining a rule comprising:

creating means for creating the rule;

associating means for associating the rule with an object class;

associating means for associating the rule with a method within the object class;

associating means for associating the rule with an occurrence of a control point within the method; and

associating means for associating the rule with another method within the object class.

61. (Original) A data processing system for defining a rule as in claim 60 wherein the occurrence of the control point within the method being before method logic.

62. (Original) A data processing system for defining a rule as in claim 60 wherein the occurrence of control point within the method being after method logic.

63. (Original) A data processing system for defining a rule as in claim 60, further comprising:

associating means for associating the rule with another object class.

64. (Cancelled)

65. (Original) A data processing system for defining a rule as in claim 60, further comprising:
associating means for associating the rule with another control point within the method of the object class.
66. (Original) A data processing system for applying a set of rules, comprising:
selecting means for selecting an object class;
selecting means for selecting a method within the object class;
invoking means for invoking the method;
processing means for processing rules associated with the method comprising:
encountering means for encountering a control point associated with the method;
determining means for determining if the control point is active; and
finding means for finding at least one rule associated with an active control point.
67. (Original) A data processing system for applying a set of rules as in claim 66, wherein the step of finding at least one rule further comprises:
accessing means for accessing a selecting algorithm associated with the active control point; and
selecting means for selecting at least one rule using the selecting algorithm.
68. (Original) A data processing system for applying a set of rules as in claim 66, where in the step of processing rules further comprises:
running means for running the at least one rule;
determining means for determining results from running the at least one rule;
accessing means for accessing a combining algorithm associated with the control point; and
combining means for combining the results using the combining algorithm.
69. (Original) A data processing system for applying a set of rules, comprising:
selecting means for selecting an object class;
selecting means for selecting a method within the object class;

- invoking means for invoking the method;
- processing means for processing rules comprising:
- encountering means for encountering a control point;
- accessing means for accessing a selecting algorithm associated with the control point; and
- selecting means for selecting at least one rule using the selecting algorithm.

70. (Previously presented) A data processing system for applying a set of rules, comprising:

- selecting means for selecting an object class;
- selecting means for selecting a method within the object class;
- invoking means for invoking the method;
- processing means for processing rules comprising:
 - encountering means for encountering a control point, the control point being a method context control point for which associated rules may be changed from a first set of rules to a second set of rules different from the first set of rules;
 - finding means for finding at least one rule associated with the control point;
 - running means for running the at least one rule;
 - determining means for determining results on the basis of running the at least one rule;
- accessing means for accessing a combining algorithm associated with the control point; and
- combining means for combining the results using the combining algorithm.

71. (Previously presented) A data processing system for applying a set of rules, comprising:

- selecting means for selecting an object class;
- selecting means for selecting a method within the object class;
- invoking means for invoking the method;
- processing means for processing rules comprising:

encountering means for encountering a first control point associated with the method;
determining means for determining if the first control point is active;
executing means for executing method logic of the method;
encountering means for encountering a second control point associated with the method;
determining means for determining if the second control point is active;
finding means for finding a set of rules associated with one of the first control point and the second control point, wherein the set of rules contains not less than zero rules.

72. (Previously presented) A data processing system for applying a set of rules, comprising:

selecting means for selecting an object class;
selecting means for selecting a method within the object class;
invoking means for invoking the method;
processing means for processing rules comprising:
encountering means for encountering a control point associated with the method, the control point being a method context control point;
finding means for finding at least one rule associated with the control point prior to executing method logic of the method;
running means for running the at least one rule;
obtaining means for obtaining results on the basis of running the at least one rule;
and
controlling means for controlling the method on the basis of the results.

73. (Original) A data processing system for applying a set of rules as in claim 72, wherein the step of controlling the method comprises:

exiting means for exiting the method.

74. (Original) A data processing system for applying a set of rules as in claim 72, wherein the step of controlling the method comprises:

executing means for executing method logic of the method.

75. (Previously presented) A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for defining an object, the instruction comprising:

instructions for defining an object;

instructions for defining a method in the object by:

instructions for defining method logic;

instructions for placing the method logic in the method;

instructions for defining at least one control point, wherein the at least one control point is a method context control point; and

instructions for placing the at least one control point in the method wherein the method logic is continuous, wherein the step of placing the at least one control point further comprises placing the at least one control in the method after the method logic.

76. (Previously presented) A computer program product for defining an object as in claim 75, wherein the instruction of placing the at least one control point further comprises placing the at least one control point in the method before the method logic.

77-78. (Cancelled)

79. (Original) A computer program product for defining an object as in claim 75, further comprises:

instructions for flagging the at least one control point on the basis of being active.

80. (Previously presented) A computer program product for defining an object as in claim 75, wherein the instruction of defining the at least one control point further comprises:

instructions for defining a rule selection algorithm associated with the at least one control point.

81. (Previously presented) A computer program product for defining an object as in claim 75, wherein the instruction of defining the at least one control point further comprises:

instructions for defining a rule result combination algorithm associated with the at least one control point.

82. (Original) A computer program product for defining an object as in claim 75, wherein the step of defining the at least one control point further comprises:

instructions for defining a rule selection algorithm for the at least one control point; and

instructions for defining a rule result combination algorithm for the at least one control point.

83. (Original) A computer program product for defining an object as in claim 75, further comprising:

instructions for associating at least one rule with the at least one control point.

84. (Previously presented) A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for defining a rule, the instruction comprising:

instructions for creating the rule;

instructions for associating the rule with an object class;

instructions for associating the rule with a method within the object class;

instructions for associating the rule with an occurrence of a control point within the method; and

instructions for associating the rule with another method within the object class.

85. (Original) A computer program product for defining a rule as in claim 84 wherein the occurrence of the control point within the method being before method logic.

86. (Original) A computer program product for defining a rule as in claim 84 wherein the occurrence of control point within the method being after method logic.

87. (Original) A computer program product for defining a rule as in claim 84, further comprising:

instructions for associating the rule with another object class.

88. (Cancelled)

89. (Original) A computer implemented process for defining a rule as in claim 84, further comprising:

instructions for associating the rule with another control point within the method of the object class.

90. (Original) A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for applying a set of rules, the instruction comprising:

instructions for selecting an object class;

instructions for selecting a method within the object class;

instructions for invoking the method;

instructions for processing rules associated with the method comprising:

instructions for encountering a control point associated with the method;

instructions for determining if the control point is active; and

instructions for finding at least one rule associated with an active control point.

91. (Original) A computer program product for applying a set of rules as in claim 90, wherein the step of finding at least one rule further comprises:

instructions for accessing a selecting algorithm associated with the active control point; and

instructions for selecting at least one rule using the selecting algorithm.

92. (Original) A computer program product for applying a set of rules as in claim 90, where in the step of processing rules further comprises:

instructions for running the at least one rule;

instructions for determining results from running the at least one rule;

instructions for accessing a combining algorithm associated with the control

point; and

instructions for combining the results using the combining algorithm.

93. (Original) A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for applying a set of rules, the instruction comprising:

instructions for selecting an object class;

instructions for selecting a method within the object class;

instructions for invoking the method;

instructions for processing rules comprising:

instructions for encountering a control point;

instructions for accessing a selecting algorithm associated with the control point;

and

instructions for selecting at least one rule using the selecting algorithm.

94. (Previously presented) A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for applying a set of rules, the instruction comprising:

instructions for selecting an object class;

instructions for selecting a method within the object class;

instructions for invoking the method;

instructions for processing rules comprising:

instructions for encountering a control point, the control point being a method context control point for which associated rules may be changed from a first set of rules to a second set of rules different from the first set of rules;

instructions for finding at least one rule associated with the control point;

instructions for running the at least one rule;

instructions for determining results on the basis of running the at least one rule;

instructions for accessing a combining algorithm associated with the control point; and

instructions for combining the results using the combining algorithm.

95. (Original) A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for applying a set of rules, the instruction comprising:

instructions for selecting an object class;

instructions for selecting a method within the object class;

instructions for invoking the method;

instructions for processing rules comprising:

instructions for encountering a first control point associated with the method;

instructions for determining if the first control point is active;

instructions for executing method logic of the method;

instructions for encountering a second control point associated with the method;

instructions for determining if the second control point is active;

instructions for finding a set of rules associated with one of the first control point and the second control point, wherein the set of rules contains not less than zero rules.

96. (Previously presented) A computer program product embodied on a computer readable medium containing instructions for a computer implemented process for applying a set of rules, the instruction comprising:

instructions for selecting an object class;

instructions for selecting a method within the object class;

instructions for invoking the method;

processing rules comprising:
instructions for encountering a control point associated with the method, the control point being a method context control point;
instructions for finding at least one rule associated with the control point prior to executing method logic of the method;
instructions for running the at least one rule;
instructions for obtaining results on the basis of running the at least one rule; and
instructions for controlling the method on the basis of the results.

97. (Original) A computer program product for applying a set of rules as in claim 96, wherein the step of controlling the method comprises:
instructions for exiting the method.

98. (Original) A computer program product for applying a set of rules as in claim 96, wherein the step of controlling the method comprises:
instructions for executing method logic of the method.